

APPARATUS AND METHOD FOR FORMING A CODING UNIT

This application claims the benefit of U.S. Provisional Application No. 60/103,081 filed on October 5, 1998 and U.S. Provisional Application No. 60/123,600 (Attorney docket number SAR 13484P) filed on March 10, 1999, which are herein incorporated by reference.

The invention relates to a data structure in the field of digital multimedia communications. More particularly, the invention relates to a formation of a data structure that improves error resilience when applied to the coding of hierarchical subband decomposed coefficients, e.g., wavelet transform coefficients.

BACKGROUND OF THE DISCLOSURE

In the field of digital multimedia communications, data streams carrying video, audio, timing and control data are packaged into various "packets". Generally, a packet is a group of binary digits that include data and control elements which are switched and transmitted as a composite whole. The data, control elements and other information are arranged in various specific formats.

Examples of such formats are disclosed in various international Standards. These standards include, but are not limited to, the Moving Picture Experts Group Standards (e.g., MPEG-1 (11172-*), MPEG-2 (13818-*) and MPEG-4 (14496-*)), H.261 and H.263. For example, MPEG defines a packet as consisting of a header followed by a number of contiguous bytes (payload) from an "elementary data stream". An elementary stream is simply a generic term for one of the coded video, coded audio or other coded bitstreams. More specifically, an MPEG-2 "transport stream" packet comprises a header, which may be four (4) or more bytes long with a payload having a maximum length of 184 bytes. Transport stream packets are part of one or more programs that are assembled into a transport stream. The transport stream is then transmitted over a channel with a particular transfer rate.

However, transmission of packets over a noisy communication channel, e.g., wireless communication, may cause corruption in the

packets received by a receiver/decoder. Furthermore, some data streams or bitstreams carry compressed data that are correlated in a manner such that partial loss of a packet may cause the receiver/decoder to discard the entire packet. Namely, compression methods are useful for representing
5 information as accurately as possible with a minimum number of bits and thus minimizing the amount of data that must be stored or transmitted. To further increase compression efficiency, some compression methods employ "significance-based" information, e.g., a significance map-value model, to indicate to a receiver/decoder the significance of the transmitted
10 information or absence of transmitted information. The "significance-based" information is often previously defined, e.g., using symbols, such that the receiver/decoder is able to decipher additional information from the transmitted information. However, the loss of compressed data such as "significance-based" information often results in substantial errors
15 when a receiver/decoder attempts to decompress or decode the corrupted data.

Additionally, another compression techniques involves the transformation of an input image into transform coefficients using hierarchical subband decomposition. For example, a useful compression
20 technique appears in the Proceedings of the International Conference on Acoustics, Speech and Signal Processing, San Francisco, Cal. March 1992, volume IV, pages 657-660, where there is disclosed a signal compression system which applies a hierarchical subband decomposition, or wavelet transform, followed by the hierarchical successive
25 approximation entropy-coded quantizer. A wavelet pyramid, also known as critically sampled quadrature-mirror filter (QMF) subband representation, is a specific type of multiresolution hierarchical subband representation of an image.

More specifically, in a hierarchical subband system, with the
30 exception of the highest frequency subbands, every coefficient at a given scale can be related to a set of coefficients at the next finer scale of similar orientation according to a structure called a wavelet tree. The coefficients at the coarsest scale will be called the parent nodes, and all coefficients

corresponding to the same spatial or temporal location at the next finer scale of similar orientation will be called child nodes.

A typical method of coding these transform coefficients is in "tree depth scan order as shown in FIG. 1, where an image is decomposed into three levels of resolution. Specifically, the wavelet coefficients are coded in tree blocks fashion, where each tree block is represented by three separate "texture units" shown with different shadings. Each texture unit is representative of a tree structure starting from the lowest or coarsest AC band to the highest or finest AC band coefficients. However, as the image size is increased, each texture unit will encompass a greater number of transform coefficients such that each texture unit is coded using more than one packet. This can cause more information loss if error occurs in one of these packets.

Namely, the loss of a portion of a texture unit, will often cause a significant error or loss of data. Therefore, there is a need in the art for an apparatus and method for formulating a data structure or coding unit, e.g., a new texture unit, to packetize such transform coefficients to improve error resilience, regardless of the packet protocol that is employed.

SUMMARY OF THE INVENTION

The present invention is an apparatus and a method for forming a data structure that improves error resilience when applied to the coding of hierarchical subband decomposed coefficients, e.g., wavelet transform coefficients. In the present invention, the data structure is referred to as a "texture unit".

In one embodiment, the texture unit is defined as comprising only those AC transform coefficients that are located in one or more rows in a single subband. For example, a single slice of transform coefficients in a HL_1 subband is collected as a texture unit and then packetized.

In a second embodiment, the texture unit is defined as comprising only those AC transform coefficients that are located in all the subbands of a decomposition level. For example, a single slice of transform

coefficients from each of the HL_3 , HH_3 , LH_3 subbands are collected as a texture unit and then packetized.

In a third embodiment, the texture unit is defined as comprising only those AC transform coefficients that are across "n" subbands, where
5 "n" is a smaller number than the total number of "N" levels of decomposition. Namely, the "depth" of a texture unit in terms of subband is limited to a fixed value of "n". This allows a larger image to be decomposed to a greater number of levels of resolution while minimizing the possibility of having a single texture unit being encoded onto more
10 than one packet. For example, transform coefficients from the HL_2 and HL_1 subbands can be collected as a texture unit and then packetized.

In a fourth embodiment, if the DC component for each of the color components (luminance (Y), C_r (U) and C_b (V)) is coded in bitplanes, then the texture unit is defined as comprising a bitplane from any color
15 components. Thus, each bitplane of the DC transform coefficients is encoded as a single texture unit.

In a fifth embodiment, the packet size varies in accordance with a subband or decomposition level of the hierarchical subband decomposed image. Namely, in the context of hierarchical subband coding, a smaller
20 packet size is employed for the lower frequency subbands and a larger packet size is employed for the higher frequency subbands.

The present coding method and data structures provide error resilience. Namely, if an error occurs in a packet or a portion thereof, the overall amount of information that is lost will be minimized. In fact, it is
25 likely that the receiver/decoder may account for the loss by applying various error recovery methods.

BRIEF DESCRIPTION OF THE DRAWINGS

The teachings of the present invention can be readily understood by
30 considering the following detailed description in conjunction with the accompanying drawings, in which:

FIG. 1 is a schematic illustration of the parent-child dependencies of subbands in an image decomposed to three levels within a wavelet tree having a plurality of texture units as used in the prior art;

FIG. 2 depicts a block diagram of a simplified packet stream system of the present invention;

FIG. 3 is a schematic illustration of a texture unit comprising AC coefficients within a single subband and its corresponding packet

5 structure;

FIG. 4 is a schematic illustration of a texture unit comprising AC coefficients across n -subbands ($n=2$) and its corresponding packet structure;

FIG. 5 is a schematic illustration of a texture unit comprising AC
10 coefficients across n -subbands ($n=3$);

FIG. 6 is a schematic illustration of a texture unit comprising of a single bitplane from the DC band;

FIG. 7 illustrates a block diagram of an encoding system and a decoding system of the present invention;

FIG. 8 is a schematic illustration of a texture unit comprising AC
15 coefficients from all subbands of a decomposition level and its corresponding packet structure; and

FIG. 9 is a detailed schematic illustration of a texture unit
20 comprising only those bits from the DC transform coefficients that form a single bitplane.

To facilitate understanding, identical reference numerals have been used, where possible, to designate identical elements that are common to the figures.

25 DETAILED DESCRIPTION

FIG. 2 depicts a block diagram of a simplified structure of a packet stream system 200 of the present invention. For illustration, a data stream such as a "transport stream" as defined in accordance with the MPEG standards is used in the packet stream system illustrated in FIG.
30 2. Although the present invention is described below using the transport stream as an example, those skilled in the art will realize that the present invention can be applied to any packet streams, e.g., an MPEG "program stream" or any other packet streams in accordance with other formats. Furthermore, although the present invention is described below using the

term "stream", it should be understood that the various operations described below may be performed on the entire stream or portion thereof.

System 200 includes an image/video encoder 220 for receiving and encoding video data 210 into an elementary video bitstream. The video encoder 220 is an encoder capable of generating hierarchical subband decomposed coefficients, e.g., wavelet coefficients with or without significance-based information. The image/video encoder 220 may be a single image encoder, e.g., a Joint Photographic Experts Group (JPEG) encoder, GIF, PICT, and the like, or an encoder for an image sequence (video), e.g., a block-based or wavelet-based image encoder operating in accordance with an MPEG standard. Throughout this disclosure the terms image sequence, images, and video are used interchangeably. In its broadest sense, the invention operates in cooperation with any form of image or image sequence encoder that would benefit from the present packet structures to provide error resilience.

One example of such an encoder is the Sarnoff Very Low Bit Rate (VLBR) encoder, which is disclosed and claimed in US patent 5,764,805 (issued on June 9, 1998), and is herein incorporated by reference. Other examples of such encoders are disclosed in U.S. patent application entitled "Apparatus And Method For Encoding Zerotrees Generated By A Wavelet-Based Coding Technique" (filed on October 24, 1996 with serial number 08/736,114), which is herein incorporated by reference.

Similarly, the system may include an audio encoder 222 for receiving and encoding audio data 212 into an elementary audio bitstream. However, those skilled in the art will realize that a plurality of image/video encoders 220_n and audio encoders 222_n can be employed to produce a plurality of elementary bitstreams. In fact, the plurality of video and audio encoders can be collectively represented by a server 225, which may employ various encoders and/or may simply contain a plurality (or a library) of stored elementary streams in various storage media. Generally, the output of such server contains interleaved program streams.

In turn, these bitstreams are sent to packetizers 230 of the present invention, where the elementary bitstreams are converted into packets.

Information for using the packets independently of the transport stream may be added when the packets are formed. Thus, non-audio/video data are allowed, but they are not shown in FIG. 2. It should be noted that although in a preferred embodiment, the present encoder and the
5 packetizer are implemented in a single module, those skilled in the art will realize that the functions performed by the encoder and the packetizer can be jointly or separately implemented as required by a particular application.

The packets are received and multiplexed by the transport stream
10 multiplexer 240 to produce a transport stream 245. Packets constructed from elementary streams that form a program (a group of "Packet Identifiers" (PIDs) with associated video and audio data) generally share a common time base. Thus, the transport stream may contain one or more programs with one or more independent time bases, where the time bases
15 are used for synchronized presentation. The time bases of different programs within a transport stream may be different.

The transport stream 245 is transmitted over a transmission
channel 250, which may further incorporate separate channel specific encoder and decoder (not shown). Next, the transport stream 245 is
20 demultiplexed and decoded by a transport stream demultiplexor 260, where the elementary streams serve as inputs to video decoder 270 and audio decoder 290, whose outputs are decoded video signals 275 and audio signals 295, respectively.

Furthermore, timing information is also extracted by the transport
25 stream demultiplexor 260 and delivered to clock control 280 for synchronizing the video and audio decoders with each other and with the channel. Synchronization of the decoders with the channel is accomplished through the use of the "Program Clock Reference" (PCR) in the transport stream. The PCR is a time stamp encoding the timing of the
30 bitstream itself and is used to derive the decoder timing.

As discussed above, the packetizer 230 organizes the bitstream from the encoder into packets for transmission. If the transmission channel 250 is noisy, the transmitted packets can be corrupted or partially lost. Although the present invention describes a method for manipulating a

bitstream to form a particular data structure or packet structure within the encoder 220, it should be understood that this operation can also be performed within the packetizer 230. As such, the implementation of the present invention is a matter of designer choice.

5 Error resilience is particularly important for packets carrying hierarchically decomposed information, i.e., hierarchical subband decomposed coefficients. Hierarchical subband decomposition provides a multi-resolution representation of an image. For example, the image is first decomposed into four subbands, LL, LH, HL, HH, each representing
10 approximately a quarter of the entire frequency band. To obtain the next coarser scale image representation, the LL band is further divided into four subbands. The process can be repeated to form a hierarchical subband pyramid. It should be understood that hierarchical subband decomposition can apply any number of subband decompositions.

15 Hierarchical subband decomposed coefficients can be packetized into units called "texture packets" for error resilience. A texture packet consists of one or more coding units, named "texture units". Namely, if the texture unit is packetized into a single packet, then the packet is referred to as a texture packet of the hierarchical subband decomposed
20 image. In the present invention, different embodiments for forming a texture unit are disclosed.

FIG. 3 is a schematic illustration of one or more texture units 320a-c comprising of a slice of AC coefficients within a single subband. It should be noted that a slice can correspond to one or more rows in a subband. The
25 texture units 320a-c are also shown disposed within a packet structure 300 separated by markers 325.

More specifically, FIG. 3 illustrates a hierarchical subband tree structure having been decomposed into three levels or scales of resolution (e.g., $l=0, 1, 2$; level 0 = HL_3, HH_3, LH_3 ; etc.), i.e., a slice in a subband is
30 defined as a texture unit. Unlike the prior art, FIG. 3 illustrates a texture unit 320a comprising only a single row of coefficients in the HL_3 subband. Similarly, FIG. 3 illustrates two other texture units 320b and 320c comprising two rows and four rows of wavelet coefficients in the HL_2 and HL_1 subbands, respectively. Namely, the texture unit comprises 2^l row(s)

of coefficients. In turn, each of these texture units is also shown as being encoded onto a texture packet 300 having a header 310, a plurality of texture units 320.

Although in the present embodiment a single texture unit is shown
5 encoded onto a single texture packet, there may be situations where it is desirable to encode more than one texture unit onto a single texture packet, e.g., small texture units. In such situation, the packet may comprise a marker 325 to demarcate the separation of two texture units 320.

10 Alternatively, FIG. 8 is a schematic illustration of one or more texture units 820a-c comprising of slices of AC coefficients within subbands of a particular decomposition level. This texture unit is referred to as having a layer-by-layer structure. Again, it should be noted that a slice can correspond to one or more rows in a subband. In turn, the
15 texture units 820a-c are also shown disposed within a packet structure 800 separated by optional markers 825.

More specifically, FIG. 8 illustrates a hierarchical subband tree structure having been decomposed into three levels or scales of resolution (e.g., $l=0, 1, 2$; level 0 = HL_3, HH_3, LH_3 ; etc.), i.e., slices in all the subbands
20 of a particular decomposition level are defined as a texture unit. Namely, FIG. 8 illustrates a texture unit 820a comprising three rows of coefficients from the HL_3, HH_3 and LH_3 subbands. Similarly, FIG. 8 illustrates two other texture units 820b and 820c comprising three sets of two rows of coefficients from the HL_2, HH_2 and LH_2 subbands and three sets of four
25 rows of coefficients from the HL_1, HH_1 and LH_1 subbands, respectively. Namely, the texture unit comprises 3×2^l row(s) of coefficients.

Alternatively, FIG. 4 is a schematic illustration of a texture unit 420 comprising of AC coefficients across "n"-subbands (e.g., $n=2$). The texture unit 420 is also shown disposed within a packet structure 400.

30 More specifically, FIG. 4 illustrates a wavelet tree structure having been decomposed into four levels or scales of resolution (e.g., $l=0, 1, 2, 3$). FIG. 4 shows an example of "depth 2" texture units, where all texture units contain the same number of coefficients. Unlike the prior art, FIG. 4 illustrates a texture unit 420a comprising coefficients across two

subbands, HL_4 and HL_3 . Similarly, FIG. 4 illustrates four other texture units 420b through 420e comprising wavelet coefficients across another two subbands, HL_2 and HL_1 . In turn, each of these texture units is shown as being encoded onto a texture packet 400 having a header 410, a texture unit 420.

Again, although in the present embodiment a single texture unit 420 is shown encoded onto a single texture packet, there may be situations where it is desirable to encode more than one texture unit onto a single texture packet, e.g., small texture units. In such situation, the packet may comprise a marker 425 to demarcate the separation of two texture units 420.

Namely, the present texture unit formation yields fixed size texture units. If an image is hierarchically subband decomposed into "N" levels, a texture unit is formed as a subtree structure of depth "n" ($n < N$) with a single coefficient as the root of the subtree, where n can vary from image to image. The formation of texture units can start either from the highest AC band or from the lowest AC band. It is possible that $(N \text{ MOD } n) \neq 0$, thus some texture units will have only $(N \text{ MOD } n)$ depth. If these texture units are too small, they can be combined with the next texture unit in coding order onto the same packet as shown in packet 400. With the new texture unit formation, the coding order of coefficients is such that all coefficients within a texture unit are coded before coding the next texture unit.

FIG. 5 shows an example of "depth 3" texture units that are formed from the highest AC bands. In this case, there are four (4) hierarchical subband decomposition levels, where the lowest AC band has only one coefficient as a texture unit that is too small, thus it is combined into the next texture unit, forming the texture unit with coefficients 1-22.

FIG. 6 is a schematic illustration of a texture unit 620 comprising of DC coefficients that are encoded in accordance with bitplanes, e.g., four bitplanes for each color components (Y, U, V). The texture unit 620 is also shown disposed within a packet structure 600.

In this embodiment, each texture unit 620 is defined as comprising only those bits from the DC transform coefficients that form a single

bitplane. For example, the DC component LL_3 for each image can be represented in three color components: luminance (Y), $C_r(U)$, and $C_b(V)$. It should be noted that the color components C_r , and C_b are typically defined as being one-fourth the size of the corresponding luminance color component.

Referring to FIG. 9, a bitplane b_n is defined as comprising those bits from the DC transform coefficients that are of common "significance". For example, FIG. 9 illustrates 16 DC coefficients for the Y color component and four corresponding DC coefficients for each of the U and V color components. The non-zero coefficient values are expressed in binary form as an example. Thus, the most significant bit (MSB) of each DC coefficient of the Y color component form one bitplane, which also is defined as a texture unit in this embodiment. In turn, the next bitplane b_{n+1} is formed from the next most significant bit (MSB) of each DC coefficient of the Y color component and so on. Therefore, FIG. 9 illustrates 12 possible bitplanes that correspond to 12 texture units 620a-1of FIG. 6. It should be noted that the number of bitplanes is dependent on the maximum magnitude of the DC coefficients. Since the DC band carries more important coefficients as compared to high frequency AC coefficients, the present embodiment of defining a texture unit for the DC coefficients based on each bitplane greatly increases error resiliency. Namely, the lost of one corrupted texture unit will only result in a partial loss of information for each DC coefficient. Thus, by using various error recovery or error concealment methods, error resiliency can be maximized for the important DC coefficients by defining each texture unit as comprising only those bits from the DC transform coefficients that form a single bitplane.

Returning to FIG. 6, although in the present embodiment a single texture unit 620 is shown encoded onto a single texture packet, there may be situations where it is desirable to encode more than one texture unit onto a single texture packet, e.g., small texture units. In such situation, the packet may comprise a marker 625 to demarcate the separation of two texture units 620. Thus, the size, shape and levels of resolution can be

selectively defined for a texture unit in accordance with a particular application using one of the above texture unit formation embodiments.

Furthermore, as noted above, the importance of the coefficients in different subbands (or frequency range) is different. In general, in hierarchical subband coding, the coefficients in the lower frequency bands are more important than the ones in higher frequency bands. Thus, instead of using constant target length for all packets, in one embodiment of the present invention, the target packet size varies in accordance with subbands and decomposition levels. Specifically, a smaller packet size is employed for the more important coefficients and a larger packet size is employed for the less important coefficients. In the context of hierarchical subband coding, a smaller packet size is employed for the lower frequency subbands and a larger packet size is employed for the higher frequency subbands. This embodiment provides greater error protection because the effect from the loss of an "important" texture packet is minimized due to its reduced packet size, i.e., losing less information for each corrupted important packet.

More specifically, the present embodiment employs a small packet size of N bits for the lowest frequency subband, e.g., LL_3 . Next, the packet size is increased to " a " x N , where " a " is greater than or equal to 1 for the subbands of the next decomposition level, e.g., HL_3 , HH_3 , and LH_3 and so on.

It should be noted that in the present invention, the coding of coefficients means the coding of the coefficient values and/or the significant symbols (e.g., if zerotree encoding is employed). Specifically, zerotree encoding is an example of an encoding method that employs wavelet transform to generate a plurality of wavelet coefficients with "coefficient significance information" in the form of a significance map. The significance map is entropy coded using a plurality of symbols: ZEROTREE ROOT (ZTR), VALUED ZEROTREE ROOT (VZTR), ISOLATED ZERO (IZ) and VALUE. A ZEROTREE ROOT symbol denotes a coefficient that is the root of a zerotree. A VALUED ZEROTREE ROOT symbol is a node where the coefficient has a non-zero value and all four children are ZEROTREE ROOTS. An ISOLATED ZERO symbol identifies a coefficient

with zero value, but with some descendant somewhere further along with a nonzero. Furthermore, if a leave has a zero value, it can also be assigned as an ISOLATED ZERO. A VALUE symbol identifies a coefficient with a non-zero value, but also with some descendant
5 somewhere further along the tree that has a non-zero value.

FIG. 7 illustrates a block diagram of an encoding system 700 and a decoding system 760 of the present invention. The encoding system 700 comprises a general purpose computer 710 and various input/output devices 720. The general purpose computer comprises a central
10 processing unit (CPU) 712, a memory 714 and an encoder/packetizer 716 for encoding and packetizing an image, video and/or audio signal.

In the preferred embodiment, the encoder/packetizer 716 is simply the video encoder 220, the audio encoder 222 and/or the packetizer 230 as discussed above in FIG. 2. It should be understood that the encoders and
15 the packetizer can be implemented jointly or separately. The encoder/packetizer 716 can be physical devices which are coupled to the CPU 712 through a communication channel. Alternatively, the encoder/packetizer 716 can be represented by a software application (or a combination of software and hardware, e.g., using application specific
20 integrated circuits (ASIC)), where the software is loaded from a storage medium, (e.g., a magnetic or optical drive or diskette) and operated by the CPU in the memory 714 of the computer. As such, the encoder/packetizer 716 of the present invention can be stored on a computer readable medium. In turn, the packets 300 through 600, which are generated by the
25 encoder/packetizer 716, can also be stored on a computer readable medium, e.g., RAM memory, magnetic or optical drive or diskette and the like.

The computer 710 can be coupled to a plurality of input and output devices 720, such as a keyboard, a mouse, an audio recorder, a camera, a
30 camcorder, a video monitor, any number of imaging devices or storage devices, including but not limited to, a tape drive, a floppy drive, a hard disk drive or a compact disk drive.

The encoding system is coupled to the decoding system via a communication channel 750. The present invention is not limited to any particular type of communication channel.

The decoding system 760 comprises a general purpose computer 770
5 and various input/output devices 780. The general purpose computer comprises a central processing unit (CPU) 772, a memory 774 and an decoder/depacketizer 776 for receiving and decoding a sequence of encoded images.

In the preferred embodiment, the decoder/depacketizer 776 is
10 simply any decoders that are complementary to the encoder/packetizer 716 as discussed above for decoding the bitstreams generated by the encoder/packetizer 716. The decoder 776 can be a physical device which is coupled to the CPU 772 through a communication channel. Alternatively, the decoder/depacketizer 776 can be represented by a software application
15 which is loaded from a storage device, e.g., a magnetic or optical disk, and resides in the memory 774 of the computer. As such, any of complementary decoders of the encoder/packetizer 716 of the present invention can be stored on a computer readable medium.

The computer 760 can be coupled to a plurality of input and output
20 devices 780, such as a keyboard, a mouse, a video monitor, or any number of devices for storing or distributing images, including but not limited to, a tape drive, a floppy drive, a hard disk drive or a compact disk drive. These input/output devices allow the computer to store and distribute the sequence of decoded video images.

25 Although various embodiments which incorporate the teachings of the present invention have been shown and described in detail herein, those skilled in the art can readily devise many other varied embodiments that still incorporate these teachings.